II. AMENDMENTS

In the claims:

1. (Original) A method for detecting device-enhanced memory modules in a system including one or more memory modules comprising system memory, wherein at least one of the one or more memory modules is a device-enhanced memory module comprising one or more embedded devices, the method comprising:

writing a data sequence to a first memory module of the one or more memory modules in the system;

reading the data sequence from the first memory module; and

identifying the first memory module as a device-enhanced memory module if the data sequence as read from the first memory module is modified from the data sequence as written to the first memory module.

2. (Original) The method of claim 1, further comprising:

identifying the first memory module as not being a device-enhanced memory module if the data sequence read from the first memory module is the same as the data sequence as written to the first memory module.

3. (Original) The method of claim 1, wherein the data sequence is modified by an embedded device comprised on the first memory module prior to said reading the data sequence from the first memory module.

4. (Original) The method of claim 1, wherein the data sequence includes one or more computer instructions that are illegal instructions for reads and writes to the system memory during normal program execution within the system, wherein the one or more illegal computer instructions

are operable to prevent the data sequence from being accidentally written to the system memory during normal operation of the system.

5. (Original) The method of claim 1, further comprising, after said identifying the first memory module as a device-enhanced memory module:

writing a plurality of data writes to the first memory module; and

the first memory module learning an error detection and correction scheme of the system from the plurality of data writes.

6. (Original) The method of claim 5, wherein each of the plurality of data writes includes a plurality of data bits and a plurality of check bits, wherein the plurality of data writes includes one data write for each of the plurality of data bits, wherein the particular data bit is set in the data write and all other data bits are not set, and wherein the check bits are set as appropriate for the data write in the error detection and correction scheme.

7. (Original) The method of claim 6, wherein said learning the error detection and correction scheme comprises:

generating check bit masks for each of the plurality of check bits using the check bits and the data bits of each of the data writes with one data bit set; and

storing the generated check bit masks;

wherein, when applying the learned error detection and correction scheme, the check bit masks are used in generating check bits for data writes to and from the first memory module.

8. (Original) The method of claim 6, wherein said learning the error detection and correction scheme comprises:

for each of the data writes with one set data bit:

storing a position of the set data bit in the plurality of data bits in a syndrome array at a location indicated by the check bits of the data write;

wherein the check bits of each of the data writes for the plurality of data bits define an error detection and correction code associated with the data bit position of the set data bit in the data write;

wherein, when applying the learned error detection and correction scheme, each of the stored data bit positions is retrievable from the syndrome array using the error detection and correction code associated with the particular data bit position.

9. (Original) The method of claim 6, wherein the plurality of data writes further includes one data write with no data bits set, wherein the check bits of the data write are set as appropriate for the data write with no data bits set in the error detection and correction scheme.

10. (Original) The method of claim 9, further comprising:

storing the check bits for the data write with no data bits set;

wherein, when applying the learned error detection and correction scheme, the stored check bits for the data write with no data bits set are used as a check bit inversion mask for data writes to and from the first memory module.

11. (Original) The method of claim 5, further comprising:

configuring error detection and correction logic of the first memory module to use the learned error detection and correction scheme;

wherein the configured error detection and correction logic is operable to provide error detection and correction for data writes to and from the first memory module using the learned error detection and correction scheme.

12. (Original) The method of claim 5, further comprising verifying the learned error detection and correction scheme for the first memory module after said learning the error detection and correction scheme.

13. (Original) The method of claim 5, further comprising:

writing a plurality of data values to the first memory module;

the first memory module generating error detection and correction data for the plurality of data values using the learned error detection and correction scheme;

examining the generated error detection and correction data; and

if the generated error detection and correction data is correct according to the learned error detection and correction scheme, indicating that said learning the error detection and correction scheme succeeded; and

if the generated error detection and correction data is not correct according to the learned error detection and correction scheme, indicating that said learning the error detection and correction scheme failed.

14. (Original) The method of claim 1, further comprising identifying interleave settings for the first memory module after said identifying the first memory module as a device-enhanced memory module, wherein the interleave settings include an interleave value and a number of cache lines used for interleaving the system memory.

15. (Original) The method of claim 14, further comprising verifying the identified interleave settings for the first memory module.

16. (Original) The method of claim 15, wherein said verifying the identified interleave settings comprises:

writing a first block of data to an input buffer on the first memory module, wherein the input buffer is used by an embedded device on the first memory module for receiving input data;

the embedded device reading the first block of data from the input buffer;

the embedded device writing the first block of data to an output buffer on the first memory module, wherein the output buffer is used by the embedded device for outputting data;

reading the first block of data from the output buffer;

examining the first block of data read from the output buffer; and

verifying the interleave settings in response to said examining the first block of data read from the output buffer.

17. (Original) The method of claim 1, further comprising:

writing a plurality of data values to each of a plurality of burst lines in a memory region of the first memory module, wherein, if the first memory module is interleaved with one or more other memory modules, a first portion of the plurality of data values are written to the memory region of the first memory module and a second portion of the plurality of data values are written to the one or more other memory modules with which the first memory module is interleaved;

reading one or more of the plurality of data values written to the memory region; and

examining the one or more data values read from the memory region to determine the interleave characteristics of the first memory module.

18. (Original) The method of claim 1, further comprising, after said identifying the first memory module as a device-enhanced memory module:

reading one or more hardware identification values from the first memory module; and

storing the one or more hardware identification values in a memory region accessible by a device driver for the first memory module.

19. (Original) The method of claim 1, further comprising, after said identifying the first memory module as a device-enhanced memory module:

writing one or more configuration values to one or more of a plurality of registers on the first memory module, wherein the configuration values are configured for use during operation of an embedded device on the first memory module;

wherein the configuration values include one or more of a sleep mode timer value, a restart timer value, an error timer value, and a driver version of a device driver for the embedded device on the first memory module.

20. (Original) The method of claim 1, wherein said writing, said reading, and said identifying are performed during system startup.

21. (Original) The method of claim 1, further comprising, after said identifying the first memory module as a device-enhanced memory module:

learning an error detection and correction scheme for the first memory module;

identifying interleave settings for the first memory module;

performing a buffer check for the first memory module; and

storing one or more hardware identification values to one or more registers on the first memory module.

22. (Original) The method of claim 21, wherein said writing, said reading, said identifying, said learning, said identifying, said performing, and said storing are performed during system startup.

23. (Original) The method of claim 1, further comprising, after said identifying the first memory module as a device-enhanced memory module, wherein said initializing includes:

initializing the first memory module to use an error detection and correction scheme of the system; and

initializing the first memory module to use a memory interleave method, if any, of the system;

wherein, after said initializing the first memory module to use an error detection and correction scheme and said initializing the first memory module to use a memory interleave method, an embedded device on the first memory module is accessible to perform one or more functions on the system.

24. (Original) The method of claim 1, further comprising:

initializing the first memory module to operate in a system memory implementation after said identifying the first memory module as a device-enhanced memory module; and

sending an end initialization sequence to the first memory module after said initializing;

wherein, after receiving the end initialization sequence, an embedded device on the first memory module is accessible to perform one or more functions on the system.

25. (Original) The method of claim 1, further comprising:

initializing the first memory module to operate in a system memory implementation after said identifying the first memory module as a device-enhanced memory module; and

wherein, after said initializing, an embedded device on the first memory module is accessible to perform one or more functions on the system.

26. (Original) The method of claim 25, further comprising:

the embedded device on the first memory module entering sleep mode after said initializing, wherein, in the sleep mode, the embedded device is not accessible to perform the one or more functions, and wherein the embedded device operates with reduced power requirements in sleep mode.

27. (Original) The method of claim 26, further comprising, after said entering sleep mode, repeating said writing, said reading, said identifying and said initializing, wherein, after said repeating, the embedded device on the first memory module is again accessible to perform the one or more functions on the system.

28. (Original) The method of claim 1, wherein an embedded device on the first memory module is a compression/decompression engine.

29. (Original) A method for detecting device-enhanced memory modules in a system including a plurality memory modules comprising system memory, wherein one or more of the plurality of memory modules are device-enhanced memory modules each comprising one or more embedded devices, the method comprising:

writing one or more data sequences including a last data sequence to each of a plurality of memory regions in the system memory;

reading at least the last data sequence from each of the plurality of memory regions in the system memory; and

identifying one or more of the plurality of memory modules as device-enhanced memory modules in response to said reading the at least the last data sequence from each of the plurality of memory regions in the system memory;

wherein a first memory module of the plurality of memory modules is identified as a device-enhanced memory module if the at least the last data sequence read from a memory region on the first memory module is different than the at least the last data sequence as written to the memory region on the first memory module.

30. (Original) The method of claim 29, wherein a second memory module is identified as not being a device-enhanced memory module if the at least the last data sequence read from a memory region on the second memory module is the same as the at least the last data sequence as written to the memory region on the second memory module.

31. (Original) The method of claim 29, wherein the plurality of memory regions includes at least one memory region on each of the plurality of memory modules comprising the system memory.

32. (Original) The method of claim 29, wherein the data sequence is modified by an embedded device comprised on each of the one or more memory modules identified as device-enhanced memory modules prior to said reading the at least the last data sequence from the one or more memory modules identified as device-enhanced memory modules.

33. (Original) The method of claim 29, further comprising, after said identifying the one or more memory modules as device-enhanced memory modules:

writing a plurality of data writes to each of the one or more memory modules identified as device-enhanced memory modules;

each of the one or more memory modules identified as device-enhanced memory modules learning an error detection and correction scheme of the system from the plurality of data writes; and

configuring error detection and correction logic of each of the one or more memory modules identified as device-enhanced memory modules in accordance with the learned error detection and correction scheme;

wherein the configured error detection and correction logic of each of the one or more memory modules identified as device-enhanced memory modules is operable to provide error detection and correction for data writes to and from the memory module in accordance with the learned error detection and correction scheme.

34. (Original) The method of claim 29, further comprising:

identifying interleave settings for each of the one or more memory modules identified as device-enhanced memory modules, wherein the interleave settings include an interleave value and a number of cache lines used for interleaving the system memory.

35. (Original) The method of claim 29, further comprising:

reading one or more hardware identification values from each of the one or more memory modules identified as device-enhanced memory modules; and

storing the one or more hardware identification values in one or more memory regions accessible by one or more device drivers for the one or more memory modules identified as device-enhanced memory modules.

36. (Original) The method of claim 29, further comprising:

for each of the one or more memory modules identified as device-enhanced memory modules:

writing one or more configuration values to one or more of a plurality of registers on the memory module, wherein the configuration values are configured for use during operation of an embedded device on the memory module;

wherein the configuration values include one or more of a sleep mode timer value, a restart timer value, an error timer value, and a driver version of a device driver for the embedded device on the memory module.

37. (Original) The method of claim 29, wherein said writing, said reading, and said identifying are performed during system startup.

38. (Original) The method of claim 29, further comprising:

initializing the one or more memory modules to operate in a system memory implementation after said identifying the one or more memory modules as device-enhanced memory modules; and

wherein, after said initializing, each of one or more embedded devices on each of the one or more memory modules is accessible to perform one or more functions on the system.

39. (Original) The method of claim 29, wherein one or more embedded devices on the one or more memory modules identified as device-enhanced memory modules are compression/decompression engines.

40 – 81. (Cancel)

82. (Original) A system comprising:

a plurality of memory modules comprising system memory, wherein at least one of the plurality of memory modules comprises one or more embedded devices;

a device driver executable within the system to:

write one or more data sequences including a last data sequence to each of a plurality of memory regions in the system memory;

read at least the last data sequence from each of the plurality of memory regions in the system memory;

identify one or more of the plurality of memory modules as device-enhanced memory modules in response to said reading the at least the last data sequence from each of the plurality of memory regions in the system memory; and

wherein a first memory module of the plurality of memory modules is identified as a device-enhanced memory module if the at least the last data sequence read from a memory region on the first memory module is different than the at least the last data sequence as written to the memory region on the first memory module.

83. (Original) The system of claim 82, wherein a second memory module is identified as not being a device-enhanced memory module if the at least the last data sequence read from a memory region on the second memory module is the same as the at least the last data sequence as written to the memory region on the second memory module.

84. (Original) The system of claim 82, wherein, after said identifying the one or more memory modules as device-enhanced memory modules, the device driver is further executable to:

write a plurality of data writes to each of the one or more memory modules identified as device-enhanced memory modules, wherein each of the one or more memory modules identified as device-enhanced memory modules is configured to learn an error detection and correction scheme of the system from the plurality of data writes;

configure error detection and correction logic of each of the one or more memory modules identified as device-enhanced memory modules in accordance with said learned error detection and correction scheme;

wherein the configured error detection and correction logic of each of the one or more memory modules identified as device-enhanced memory modules is operable to provide error detection and correction for data writes to and from the memory module in accordance with the learned error detection and correction scheme.

85. (Original) The system of claim 82, wherein, after said identifying the one or more memory modules as device-enhanced memory modules, the device driver is further executable to:

identify interleave settings for each of the one or more memory modules identified as device-enhanced memory modules, wherein the interleave settings include an interleave value and a number of cache lines used for interleaving the system memory.

86. (Original) The system of claim 82, wherein, after said identifying the one or more memory modules as device-enhanced memory modules, the device driver is further executable to:

read one or more hardware identification values from each of the one or more memory modules identified as device-enhanced memory modules; and

store the one or more hardware identification values in one or more memory regions accessible by the device driver.

87. (Original) The system of claim 82, wherein, after said identifying the one or more memory modules as device-enhanced memory modules, the device driver is further executable to:

for each of the one or more memory modules identified as device-enhanced memory modules:

write one or more configuration values to one or more of a plurality of registers on the memory module, wherein the configuration values are configured for use during operation of an embedded device on the memory module;

wherein the configuration values include one or more of a sleep mode timer value, a restart timer value, an error timer value, and a driver version of a device driver for the embedded device on the memory module.

88. (Original) The system of claim 82, wherein the device driver is configured to perform said writing, said reading, and said identifying during system startup.

89. (Original) The system of claim 82, wherein the device driver is further executable to:

initialize the one or more memory modules to operate in a system memory implementation of the system after said identifying the one or more memory modules as device-enhanced memory modules; and

wherein, after said initializing, each of one or more embedded devices on each of the one or more memory modules is accessible to perform one or more functions on the system.

90. (Original) The method of claim 82, wherein one or more embedded devices on the one or more memory modules identified as device-enhanced memory modules are compression/decompression engines.

91 - 105. (Cancel)

106. (Original) A carrier medium comprising program instructions for detecting device-enhanced memory modules in a system including one or more memory modules comprising system memory, wherein at least one of the one or more memory modules is a device-enhanced memory module comprising one or more embedded devices, wherein the program instructions are computer-executable to implement:

writing a data sequence to a first memory module of the one or more memory modules in the system;

reading the data sequence from the first memory module;

identifying the first memory module as a device-enhanced memory module if the data sequence as read from the first memory module is modified from the data sequence as written to the first memory module; and

identifying the first memory module as not being a device-enhanced memory module if the data sequence read from the first memory module is the same as the data sequence as written to the first memory module.

107 - 108. (Cancel)

III. CONCLUSION

In view of the amendments set forth herein, this application is believed to be in condition for allowance and a notice to that effect is solicited. Nonetheless, should any issues remain that might be subject to resolution through a telephonic interview, the Examiner is requested to telephone the undersigned.

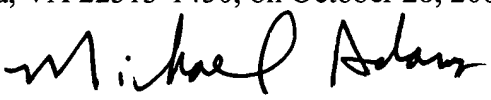
Respectfully submitted,

Michael P. Adams
Attorney for Applicant(s)
Reg. No. 34,763

P.O. Box 50784
Dallas, TX 75201
512.370.2858

CERTIFICATION UNDER 37 C.F.R. § 1.8

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to COMMISSIONER FOR PATENTS, P.O. Box 1450, Alexandria, VA 22313-1450, on October 28, 2004.



Signature

Austin_1\264976\1
40532-P008US 10/28/2004